CLAIMS

1. A set of virtual machine instructions suitable for execution in a virtual machine to load constant values on an execution stack, the set of virtual machine instructions representing a number of corresponding Java Bytecode executable instructions that are also suitable for execution in the virtual machine to load constant values on an execution stack,

wherein the set of the virtual machine instructions consists of a number of virtual machine instructions that is less than the number of the corresponding Java Bytecode executable instructions, and

wherein every one of the corresponding Java Bytecode executable instructions can be represented by at least one of the virtual machine instructions in the virtual machine instruction set.

2. A set of virtual machine instructions as recited in claim 1,

wherein the set consists of a first, a second, and a third instruction, the first instruction suitable for pushing one byte values on the execution stack, the second instruction suitable for pushing 4 byte values on the execution stack, and the third instruction suitable for pushing 8 byte values on the execution stack.

3. A set of virtual machine load constant instructions as recited in claim 1,

wherein the first instruction includes a code portion and a data portion which are both represented in a code stream in the virtual machine, and

wherein the second instruction includes a code portion and a data portion which are respectively represented in a code stream and in a data stream in the virtual machine.

4. A set of virtual machine instructions as recited in claim 3,

wherein the set is suitable to load N byte constant values on the execution stack, and

wherein N is a positive integer.

5. A set of virtual machine instructions suitable for execution in a virtual machine to store local variables onto an execution stack, the set of virtual machine instructions representing a number of corresponding Java Bytecode executable instructions that are also suitable for execution in the virtual machine to store local variables onto an execution stack,

wherein the set of virtual machine instructions consists of a number of virtual machine instructions that is less than the number of the corresponding Java Bytecode executable instructions, and

wherein every one of the corresponding Java Bytecode executable instructions can be represented by at least one of the virtual machine instructions in the virtual machine instruction set.

6. A set of virtual machine instructions as recited in claim 5,

wherein the set of virtual machine instructions consists of a first instruction and a second instruction, the first instruction being suitable for storing 4 byte local variables onto the execution stack, and the second instruction being suitable for storing 8 byte local variables onto the execution stack.

7. A set of virtual machine instructions as recited in claim 5, wherein the first or the second instruction includes a code portion and a data portion which are respectively represented in a code stream and in a data stream in the virtual machine.

8. A set of virtual machine instructions as recited in claim 5,

wherein the set of set of virtual machine instructions is suitable to store N byte local variables on the execution stack, and

wherein N is a positive integer.

9. A virtual machine instruction suitable for execution in a virtual machine to load values from arrays on an execution stack, the virtual machine instruction representing two or more Java Bytecode executable instructions that are also suitable for loading values from arrays on the execution stack.

10. A virtual machine instruction as recited in claim 9, wherein the arrays can be an array of 1 byte values, or an array of 2 byte values, or an array of 4 byte values, or an array of 8 byte values.

11. A virtual machine instruction as recited in claim 9, wherein a header of an array is read to determine the type of the array.

12. A virtual machine instruction suitable for execution in a virtual machine to store values located on an execution stack into arrays, the virtual machine instruction representing two or more Java Bytecode executable instructions that are also suitable for storing values located on an execution stack into an array.

13. A virtual machine instruction as recited in claim 12, wherein the arrays can be an array of 1 byte values, or an array of 2 byte values, or an array of 4 byte values, or an array of 8 byte values.

14. A virtual machine load array instruction as recited in claim 12, wherein a header of an array is read to determine the type of the array.

15. A virtual machine instruction suitable for execution in a virtual machine to duplicate values stored in an execution stack on top of the execution stack, the virtual machine instruction representing two or more Java Bytecode executable instructions that are also suitable for duplicating values stored in the execution stack on top of the execution stack.

16. A virtual machine instruction as recited in claim 15, wherein values that can be duplicated on the execution stack are not limited to values that are within first, second, and third positions from the top of the stack.

17. A virtual machine instruction as recited in claim 15, wherein the values duplicated on top of the stack can be 4 byte values or 8 byte values.

18. A virtual machine instruction suitable for execution in a virtual machine to duplicate values stored in an execution stack on top of the execution stack, wherein the virtual machine instruction has a parameter associated with it to indicate which value stored in the execution stack should be duplicated on the top of the stack.

19. A virtual machine instruction as recited in claim 18, wherein values that can be duplicated are not limited to values that are within first, second, and third positions from the top of the execution stack.

20. A virtual machine instruction as recited in claim 18, wherein the values can be 4 byte values or 8 byte values.

21. A virtual machine instruction suitable for execution in a virtual machine to return values by placing them on top of an execution stack, the virtual machine instruction representing two or more Java Bytecode executable instructions that are also suitable for returning values by placing them on top of the execution stack.

22. A virtual machine instruction as recited in claim 21, wherein the values returned can be 4 byte values or 8 byte values.

23. A virtual machine instruction suitable for execution in a virtual machine to instantiate Java objects and arrays, the virtual machine instruction

representing two or more Java Bytecode executable instructions that are also suitable for instantiation of Java objects or arrays.

24. A virtual machine instruction as recited in claim 23, wherein the instantiation of Java objects and arrays are performed by determining the type of the object or array based on a parameter that is associated with the object or array.